Author(s): Liu, Liqing; Chang, Zheng; Guo, Xijuan; Mao, Shiwen; Ristaniemi, Tapani

Title: Multi-objective Optimization for Computation Offloading in Fog Computing

Year: 2018

Version:

Please cite the original version:

Liu, L., Chang, Z., Guo, X., Mao, S., & Ristaniemi, T. (2018). Multi-objective Optimization for Computation Offloading in Fog Computing. IEEE Internet of Things Journal, 5(1), 283-294. https://doi.org/10.1109/JIOT.2017.2780236

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

Multi-objective Optimization for Computation Offloading in Fog Computing

Liqing Liu, Zheng Chang, *Senior Member, IEEE,* Xijuan Guo, Shiwen Mao, *Senior Member, IEEE*, Tapani Ristaniemi, *Senior Member, IEEE*

***Abstract*—Fog computing system is an emergent architecture for providing computing, storage, control, and networking capabilities for realizing Internet-of-Things (IoT). In the fog computing system, the mobile devices (MDs) can offload its data or computational expensive tasks to the fog node within its proximity, instead of distant cloud. Although offloading can reduce energy consumption at the MDs, it may also incur a larger execution delay including transmission time between the MDs and the fog/cloud servers, and waiting and execution time at the servers. Therefore, how to balance the energy consumption and delay performance is of research importance. Moreover, based on the energy consumption and delay, how to design a cost model for the MDs to enjoy the fog and cloud services is also important. In this paper, we utilize queuing theory to bring a thorough study on the energy consumption, execution delay and payment cost of offloading processes in a fog computing system. Specifically, three queuing models are applied respectively to the MD, fog and cloud centers, and the data rate and power consumption of the wireless link are explicitly considered. Based on the theoretical analysis, a multi-objective optimization problem is formulated with a joint objective to minimize the energy consumption, execution delay and payment cost by finding the optimal offloading probability and transmit power for each MD. Extensive simulation studies are conducted to demonstrate the effectiveness of the proposed scheme and the superior performance over several existed schemes are observed.**

***Index Terms*—fog computing; cloud computing; energy consumption; execution delay; cost; offloading probability; power allocation**

I. Introduction

A. Background and Motivation

With the rapid development of ICT industry, mobile devices (MDs) have become an indispensable part of our daily life as they can provide convenient communications almost anytime and anywhere. The mobile application markets are also boosted by the advanced mobile technologies and high data rate wireless networks. However, due to the restrictions of the MDs on size, weight, battery life, and heat dissipation, the gap between the capability of limited computing resources and demand for executing complex applications is gradually increasing [1]. Many computational-intensive and delay-intensive mobile applications have poor performance when they are executed on devices, especially for Internet-of-Things (IoT) devices which are particularly limited with transmission power, storage, and computing resources [2].

Recent study shows that mobile cloud computing (MCC) technology provides a promising opportunity to overcome the limitation of hardware and obtain energy saving for the MDs by offloading the computational-intensive tasks to the cloud for execution [3], [4], [5], [6]. After execution in the cloud, the final results are returned back to the MDs. By such, MCC is able to efficiently overcome the limitations of processing capabilities or battery capacity of the MD. To date, several types of mobile cloud architectures are categorized [7], such as the traditional central cloud [6], [8], ad hoc mobile cloud [9], [10], cloudlet [11], [12], [13], [14], etc. The traditional central cloud (such as Amazon EC2 cloud, Microsoft Windows Azure or Rackspace) can provide huge storage, rich computational resources, as well as good security. By offloading different components of mobile applications to the cloud server, the performance of mobile applications can be greatly improved and the energy consumption of the MDs can be significantly reduced [4], [6]. However, it is worth mentioning that the traditional central cloud is usually remotely located and far away from their users. Thus, for latency-sensitive mobile applications, such as high quality video streaming, mobile gaming and so on, offloading to the distant central cloud may not be a perfect solution. Therefore, the traditional centralized cloud is encountering growing challenges, for the future mobile networks, especially for the emerging IoT paradigm.

To overcome these disadvantages, fog computing, also known as "cloud at the edge," [4], [15] emerges as an alternative proximity solution to provide pervasive and agile computation services for the MDs at anytime and anywhere and support future cloud services and applications, especially to the Internet-of-Things (IoT) applications with strict requirement of latency and high resilience [16], [17]. As a novel MCC paradigm, fog computing can provide computing resources at the edge of radio access networks (RAN) [4]. In this case, the need for interactive response between fog computing and cloud center can be met by fiber transmission from the network edge to the central cloud computing infrastructures with low-latency. The idea of using fog computing brings both computational and radio resource more closer to the MDs, thus

L. Liu and X. Guo are with Colleage of Information Science and Engineering, Yanshan University, 066004 Qinhuangdao, China. E-mail: liuliqing_yanyan@163.com, xjguo@ysu.edu.cn. Z. Chang and T. Ristaniemi are with Department of Mathematical Information Technology, University of Jyväskylä, P.O.Box 35, FIN-40014 Jyväskylä Finland. E-mail: {zheng.chang, tapani.ristaniemi }@jyu.fi. S. Mao is with Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201, USA. E-mail: smao@ieee.org. Corresponding author is Zheng Chang

This work is partly supported by the Academy of Finland (Decision number 284748) and Hebei NSF (F2016203383, E2017203351). Shiwen Mao's work was supported in part by the NSF under Grant CNS-1702957, and by the Wireless Engineering Research and Education Center (WEREC) at Auburn University, Auburn, AL, USA.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

improving scalability from both computation and radio aspects [18], [19]. However, it can be noticed that the computational resource in the fog node cannot be treated as sufficiently as that in the traditional central cloud, as it is usually targeted to serve a small portion of users.

With MCC, the mobile requests from the IoT applications can be locally executed or offloaded to the cloud for processing. However, offloading may incur additional delay and generate related cost for enjoying the cloud service. Specifically, to minimize the delay performance and cost of mobile requests, one may run the tasks locally at the MDs as no additional costs for communication time, waiting delay at the cloud and resource utilization are incurred. However, running too many requests locally may consume large amount of energy, thus shorten the lifetime of the MDs. On the contrary, offloading the requests to the cloud can save the energy at the MDs, but it unavoidably incurs the corresponding delay including waiting time at the server and the communication time between the cloud and the MDs, and the payment cost for utilizing the resource in cloud server. Thus, the tradeoff among energy consumption, delay performance and payment cost for the MDs needs to be addressed [20], [21]. In this paper, our aim is to investigate such a tradeoff in a heterogeneous fog computing environment and propose optimal offloading and power allocation policies.

B. Contributions

In this paper, we investigate the problem of joint energy consumption, delay and payment cost (E&D&P) minimization for the MDs in a fog computing heterogeneous network. The main contribution of this paper is summarized as follows:

1) A fog-based mobile cloud computing system is investigated. Different queue models are applied to different network elements in order to provide in-depth study on energy consumption and delay performance, e.g., the queues at the MD is considered as a $M/M/1$ queue, the one at the fog node is considered as a $M/M/c$ queue with a defined maximum request rate, and the one at the central cloud is considered as a $M/M/\infty$ queue. Such a fog computing system is rarely studied in the previous works about MCC. In particular, both wireless transmission and computing capabilities are explicitly and jointly considered when modelling the energy consumption, delay performance and payment cost.
2) We present a joint E&D&P optimization problem, including the energy consumption and delay in local execution process, computational task transmission process, fog execution and transmission process, and central cloud execution and transmission process, together with the payment cost, which can thoroughly complement the existing analysis of the fog computing system.
3) A multi-objective optimization problem is formulated, which involves minimizing the energy consumption, delay and payment cost by finding the optimal offloading probability and transmit power. Using the scalarization method, we are able to transform the multi-objective optimization problem into a single-objective optimization problem. Interior Point Method (IPM) is then applied to address transformed optimization problem. The proposed IPM-based algorithm can reduce the accumulated error and improve the calculation accuracy during the iteration process effectively.
4) Extensive simulation studies are conducted to evaluate the effectiveness of the proposed schemes. It is shown that our scheme can find the optimal offloading probability and transmit power, and to achieve the E&D&P minimization.

The reminder of this paper is organized as follows. We briefly overview the recent related works in Section II. In Section III the system model is introduced and the joint E&D&P optimization problem is presented. In Section IV we propose a scalarization and IPM based algorithm to solve the formulated problem. The simulation results are presented to verify the proposed schemes in Section V. Finally, we conclude our work in Section VI.

II. Related Work

In the MCC, offloading study is an attractive yet challenging topic, which involves making decisions regarding where to run the mobile requests and how to allocate computing resources. In [2], the authors summarize the opportunities and challenges of fog, focusing primarily in the networking context of IoT. As an architecture, fog supports a growing variety of applications, including those in the Internet of Things (IoT), fifth-generation (5G) wireless systems, and embedded artificial intelligence (AI). In [4], the authors introduce the definition of edge computing, followed by several case studies, ranging from cloud offloading to smart home and city, as well as collaborative edge to materialize the concept of edge computing. In [5], the authors consider a mobile computation offloading problem where multiple mobile services in workflows be invoked to fulfill their complex requirements and the decisions be made on whether the services of a workflow should be offloaded. In [6], the authors presents a quantitative study on the energy-traffic tradeoff problem from the perspective of entire Wireless Local Area Network (WLAN). In [8], the authors review first a series of offloading mechanisms and then to provide a mathematical formulation of these problems aimed at optimizing the communication and computation resources jointly, posing a strict attention to latency and energy constraints. Wherever possible, the authors also try to emphasize those features of 5G systems that can help meet the strict latency constraints while keeping the energy consumption at a minimum level. The authors of [10] study the problem that nearby mobile devices can efficiently be utilized as a crowd-powered resource cloud to complement the remote clouds and present a work-sharing mode using an adaptation of the well-known work stealing method to load balance independent jobs among heterogeneous mobile nodes. Vehicular cloud is a practical application of ad hoc mobile cloud. In [11], the authors develop a Markov decision process (MDP)-based optimal offloading algorithm for the mobile user in an intermittently connected cloudlet system, considering the users' local load and availability of cloudlets. The authors of [12] consider a multi-resource

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

allocation problem in the cloudlet environment for resource-intensive and latency-sensitive mobile applications.

Meanwhile, the average transmission delay between the MDs and central cloud can be relatively long. To address such a problem, cloudlet/fog deployed in the vicinities of users has gained recognition as an alternative offloading destination due to its short response time and relatively large capability. In [13], the authors study different cloudlet placement problems in a large scale Wireless Metropolitan Area Network (WMAN) consisting of many wireless access points (APs), with the objective to minimize the average access delay between mobile users and the cloudlet. In [14], the authors design a threshold-based policy to improve the QoS of MCC by cooperation of the local cloud and Internet cloud resources, which takes the advantages of low latency of the local cloud and abundant computational resources of the Internet cloud simultaneously. Fog computing is a new concept emerged in recent years and provides pervasive and agile computation augmenting services for the MDs with short delay [15]–[19]. The article of [16] introduces a layered fog-to-cloud architecture and its benefits, as well as the arising open and research challenges. In [17], the authors study the multi-user computation offloading problem for fog computing in a multi-channel wireless interference environment and show that the problem is NP-hard to compute a centralized optimal solution, and adopt a game theoretic approach for achieving efficient computation offloading in a distributed manner. In [18], the tradeoff between power consumption and transmission delay in the fog-cloud computing system is investigated. The authors formulate a workload allocation problem which suggests the optimal workload allocations between fog and cloud toward the minimal power consumption with the constrained service delay and solving it using an approximate approach by decomposing the primal problem into three subproblems of corresponding subsystems. In [19], the authors formulate the offloading problem as the joint optimization of the radio resources (the transmit precoding matrices of the mobile users) and the computational resources (the CPU cycles/second assigned by the fog to each mobile user), in order to minimize the overall users' energy consumption, while meeting latency constraints.

It can be found that some of the aforementioned literatures take the energy consumption, delay performance, or cost for resource usage individually into account when designing the offloading schemes. However, to date, the problem of jointly optimizing these three goals in a fog computing system has not been well addressed. Moreover, most of the previous works consider transmit power fixed, which is too simplistic and inconsistent with the reality. In addition, many works consider the cloud or fog are with infinite computing servers or capabilities whereas the reality is against it. Therefore, in this paper, we first thoroughly analyze the related energy consumption, delay performance, and cost models, and then formulate a joint E&D&P optimization to find the optimal offloading and power allocation solutions.

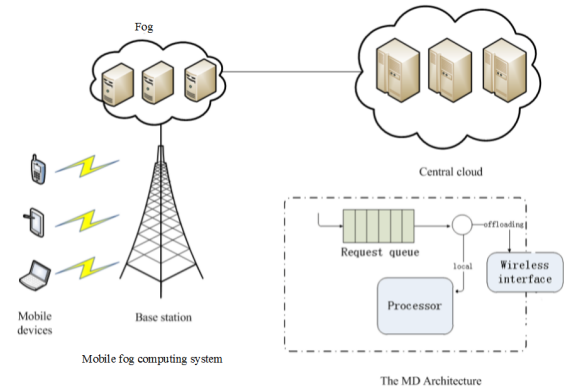


Fig. 1. The model of the fog computing system

TABLE I
NOTATIONS

Notations	Meanings
N	the number of MDs in the system
λ_i	the average request arrival rate of the MD i
p_i^C	the offloading probability of the MD i
u_i^M	the computing capability of the MD i
l_i^M	the normalized workload of the MD i
κ_i	the locally execution power of MD i
θ_i	the computation input data size in each request of MD i
W	the channel bandwidth
P_i	the transmission power of the MD i
P_i^{max}	the maximum transmission power of MD i
h_i	the channel gain between the MD i and the base station
ω_i	the background interference power
c	the number of servers in the fog node
u^F	the service rate in the fog node
u_b^F	the sending rate of the fog node
T^o	the fixed delay from fog to the central cloud
u^{CC}	the service rate of the central cloud
u_b^{CC}	the sending rate of the central cloud
$\tilde{E}$	the expected energy consumption of MDs in the system
$\tilde{T}$	the expected delay performance of MDs in the system
$\tilde{M}$	the expected payment cost of MDs in the system

III. System Model and Problem Formulation

A. System Model

As shown in Fig. 1, we assume that the considered system consists of N MDs, a fog node, and a distant central cloud. The MD can connect with the fog/cloud via the deployed base station (BS). The set of MD is denoted as $\mathcal{N}$. Each MD executes an application and generates a series of service requests. In this paper, we consider the traffic model at the MD as an $M/M/1$ queue [22], the one in the fog node as an $M/M/c$ queue [13] and the one at the central cloud as an $M/M/\infty$ queue [22]. For each MD, it can offload a portion or the whole of its requests to the fog node through the wireless channel, where the transmission suffers from interference generated by other MDs. If the total request rate is less than the maximum accepted rate of the fog node, then all the offloaded requests will be processed in the fog node. Otherwise, the fog node will further offload the overloaded requests to the central cloud for execution.

We assume that the requests generated from MD $i, i \in \mathcal{N}$, follow a Poisson process with an average arrival rate of λ_i [13]. The requests are assumed to be computationally intensive,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

mutually independent, and can be executed either locally in the MD or remotely on the fog node and the central cloud via computation offloading. Each request generated from the MD i is of data size θ_i. The MD chooses to offload the service request with a probability $p_i^C, 0 \leq p_i^C \leq 1$. Accordingly, the service requests which are offloaded to the cloud follow a Poisson process with an average rate of $p_i^C \lambda_i$ and it is denoted as the offloading rate. The service requests that are processed locally also follow a Poisson process with average rate of $\left(1-p_i^C\right)\lambda_i$, and it is called as local execution rate. We can observe that when the value of p_i^C becomes larger, more requests are delivered to the fog node or the central cloud while the less requests are processed locally. The key notations are summarized in Table 1.

Let u_i^M denotes the computing capability of MD i. Additionally we assume that l_i^M denotes the normalized workload on the MD i which represents the percentages of CPU that have been occupied. $l_i^M = 0$ indicates that the CPU is totally idle. When considering a $M/M/1$ queue at the MD, the average response time T_i^M for locally processing requests at MD i is expressed as follows [22]:

$$T_i^M\left(p_i^C\right) = \frac{1}{u_i^M\left(1-l_i^M\right) - \left(1-p_i^C\right)\lambda_i}. \tag{1}$$

When MD i transmits the data to the fog node, with the consideration of the interference caused by other MDs, we can obtain the uplink data rate for computation offloading of MD i as follows:

$$R_i = W\log_2\left(1 + \frac{P_i h_i}{\omega_0 + \sum_{j\in N, j\neq i} P_j h_j}\right), \tag{2}$$

where W is the channel bandwidth and P_i is the transmission power of the MD i. Additionally, $0 < P_i < P_i^{max}$, where P_i^{max} is the maximum transmit power of MD i. h_i is the channel gain between MD i and the BS. ω_0 denotes the noise power. Note that (2) is the the worst case that all MDs are transmitting simultaneously without any coordination. From (2), we can obtain the transmission time of MD i for offloading the data from MD i as follows:

$$T_i^t\left(p_i^C, P_i\right) = \frac{p_i^C \lambda_i \theta_i}{R_i}. \tag{3}$$

As one can observe, the energy consumption of MD i comprises of two parts: (1) energy consumption of the MD for local service request processing; (2) energy consumption for transmitting data to the BS. The energy consumption $E_i^M\left(p_i^C\right)$ for locally executing the requests for MD i can be given as follows:

$$E_i^M\left(p_i^C\right) = \kappa_i T_i^M\left(p_i^C\right) = \kappa_i \frac{1}{u_i^M\left(1-l_i^M\right) - \left(1-p_i^C\right)\lambda_i}, \tag{4}$$

where κ_i is the energy coefficient denoting the locally executing power of MD i, which is related to the intrinsic nature of the MDs. For the sake of simplicity, we assume κ_i is constant during the waiting time and computation process.

We denote the energy consumption for transmitting the requests from the MD to the BS is $E_i^S\left(p_i\right)$, which can be given as follows [19]:

$$\begin{aligned} E_i^S\left(p_i^C, P_i\right) &= P_i T_i^t\left(p_i^C\right) = P_i \frac{p_i^C \lambda_i \theta_i}{R_i} \\ &= \frac{P_i p_i^C \lambda_i \theta_i}{W\log_2\left(1 + \frac{P_i h_i}{\omega_0 + \sum_{j\in N, j\neq i} P_j h_j}\right)}. \end{aligned} \tag{5}$$

It can be noticed that the computing resource of the fog node may be adequate for running several mobile requests simultaneously, but insufficient for executing too many requests. The central cloud, on the other hand, has sufficient computing resources. So it can be considered to be always available as long as the users purchase the service. Therefore, if the fog node is overloaded, the overloaded requests will be further offloaded to the central cloud.

Accordingly, we assume that there are c homogeneous servers deployed in the fog node. The service rate for each server is denoted as u^F. The maximum workload of the fog node is capped at a maximum request rate denoted as $\lambda_{\max}^F$. The purpose of defining $\lambda_{\max}^F$ for the fog node is to avoid the excessive queueing delay when the fog node servers are heavily loaded. The requests from different MDs in the system are pooled together with a total rate λ_{Total}^M. According to the properties of the Poisson process, λ_{Total}^M is given as follows:

$$\lambda_{Total}^M = \sum_{i=1}^{N} \lambda_i p_i^C. \tag{6}$$

Then the fraction of the requests ψ^F that the fog node can process is given as:

$$\psi^F = \begin{cases} 1, & \lambda_{\max}^F \geq \lambda_{Total}^M; \\ \frac{\lambda_{\max}^F}{\lambda_{Total}^M}, & \lambda_{\max}^F < \lambda_{Total}^M. \end{cases} \tag{7}$$

Correspondingly, the actual execution rate at the fog node can be expressed as:

$$\lambda_p^F = \psi^F \lambda_{Total}^M = \begin{cases} \lambda_{Total}^M, & \lambda_{\max}^F \geq \lambda_{Total}^M; \\ \lambda_{\max}^F, & \lambda_{\max}^F < \lambda_{Total}^M. \end{cases} \tag{8}$$

To this end, based on the analysis of $M/M/c$ queue at the fog node and Erlang's Formula [23], we define

$$\rho^F = \frac{\lambda_p^F}{cu^F}. \tag{9}$$

Therefore, the average waiting time of each request at the fog node, which contains the waiting time and execution time, is denoted as follows [13], [23]:

$$T_{wait}^F\left(\lambda_p^F\right) = \frac{C\left(c, \rho^F\right)}{cu^F - \lambda_p^F} + \frac{1}{u^F}, \tag{10}$$

where

$$C\left(c, \rho^F\right) = \frac{\left(\frac{\left(c\rho^F\right)}{c!}\right)\left(\frac{1}{1-\rho^F}\right)}{\sum_{k=0}^{c-1} \frac{\left(c\rho^F\right)^k}{k!} + \left(\frac{\left(c\rho^F\right)}{c!}\right)\left(\frac{1}{1-\rho^F}\right)}. \tag{11}$$

Assuming u_b^F is the transmission data rate of the fog node, we can obtain the expected time T_b^F for the execution results waiting in the fog node before they are completely delivered out as follows:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

$$T_b^F\left(\lambda_p^F\right)=\frac{1}{u_b^F-\lambda_p^F}. \tag{12}$$

If the fog node cannot process all the requests due to the limitation of computational resources, overloaded requests are transmitted to the central cloud through wired connection. Accordingly, we assume that the transmission of those requests to the central cloud incurs a fixed time delay T^O. As the central cloud has sufficient computing resources to process these requests, the queuing time of the requests in the central cloud can be negligible. The queue model at the central cloud is considered as $M/M/\infty$ with the service rate u^{CC}, which is usually faster than the fog node service rate u^F. Then, the waiting time T_{wait}^{CC} of the overloaded requests, which includes the transmission time from the fog node to the central cloud, and the execution time at the central cloud can be presented as follows:

$$T_{wait}^{CC}=T^O+\frac{1}{u^{CC}}. \tag{13}$$

After processing the tasks, the central cloud will transmit the results to the fog node, and the fog node would send the results to the MDs, since the central cloud might not know the IP address of the MD. Then the expected time T_b^{CC} for the results waiting in the cloud before they are completely sent out is denoted as

$$T_b^{CC}\left(p_i^C\right)=\frac{1}{u_b^{CC}-\left(\lambda_{Total}^M-\lambda_p^F\right)}. \tag{14}$$

The time and energy consumption for the MD to receive the results can be ignored, due to the fact that for many applications (e.g., face recognition), the size of the computation outcome in general is much smaller than that of input data [5], [17]. From (4) and (5), we can obtain the energy consumption of MD i as follows:

$$E_i\left(p_i^C,P_i\right)=\left(1-p_i^C\right)E_i^M\left(p_i^C,P_i\right)+p_i^C E_i^S\left(p_i^C,P_i\right). \tag{15}$$

From (1), (3), (10), (12), (13) and (14), we can obtain the execution time of MD i, which is denoted as follows

$$\begin{aligned}T_i\left(p_i^C,P_i\right)=&\left(1-p_i^C\right)T_i^M\left(p_i^C\right)+p_i^C T_i^t\left(p_i^C,P_i\right)\\&+p_i^C\psi^F\left(T_{wait}^F+T_b^F\right)\\&+p_i^C\left(1-\psi^F\right)\left(T_{wait}^{CC}+T_b^{CC}\right).\end{aligned} \tag{16}$$

Correspondingly, the average energy consumption and execution time of all MDs in the system are given in (17) and (18).

In addition, the MD has to pay for the resource they used in the fog node or the central cloud. We assume that the unit cost for the fog node is r^F and that for the central cloud is r^{CC}. In general, $r^{CC}>r^F$ as the central cloud has a number of powerful servers that need a lot of resources to maintain and it can also encourage the use of fog computing. We also assume that the cost is related to the use of resources, e.g., execution rate. Through the above assumptions, we can compute the average cost of the MDs as follows:

$$M\left(p_i^C\right)=\frac{1}{N}\left\{r^F\lambda_p^F\left(p_i^C\right)+r^{CC}\left[\lambda_{Total}^M\left(p_i^C\right)-\lambda_p^F\left(p_i^C\right)\right]\right\}. \tag{19}$$

B. Problem Formulation

To this end, with above analytic results on the expected energy consumption, execution delay and payment cost performance, we are able to formulate the joint E&D&P minimization problem. The problem can be considered as a multi-objective optimization which involves minimizing energy consumption, execution delay and cost, as follows:

$$\textbf{P1}:\quad \min_{\{p_i^C,P_i\}}\left\{E\left(p_i^C,P_i\right),T\left(p_i^C,P_i\right),M\left(p_i^C\right)\right\}, \tag{20}$$

subject to

$$\left(1-p_i^C\right)\lambda_i<u_i^M\left(1-l_i^M\right), \tag{21}$$

$$\lambda_p^F<cu^F, \tag{22}$$

$$\lambda_p^F<u_b^F, \tag{23}$$

$$\lambda_{Total}^M-\lambda_p^F<u_b^{CC}, \tag{24}$$

$$0<P_i<P_i^{max}\qquad\forall i\in\mathcal{N}, \tag{25}$$

$$0\le p_i^C\le 1\qquad\forall i\in\mathcal{N}. \tag{26}$$

Constraints (21), (22), (23), and (24) are derived from (1), (10), (12), and (14) respectively. (21) enforces that the request arrival rate of local execution should not exceed the MD's processing rate. (22) enforces that the actual processing rate at the fog node should not exceed the service rate of the fog node. (23) makes sure that the actual processing rate at the fog node should not exceed the transmission rate of the fog node and (24) ensures that the requests arrival rate at the central cloud should not exceed the transmission rate of the central cloud.

It can be noticed that the formulated problem is a multi-objective nonlinear optimization problem with various constraints. As discussed in [24], in general, there are two kinds of algorithms to solve the multi-objective optimization problems, which are traditional optimization algorithm and intelligent optimization algorithm. The traditional optimization methods include weighted method, constraint method, linear programming method and so on. In the use of the weighted method, the first goal is to obtain a dimensionless process for each objective function. Therefore, we assume that the MDs in the system have an expected maximum energy consumption, execution delay and payment cost, which are denoted as $\tilde{E}$, $\tilde{T}$, $\tilde{M}$ respectively and they are all constants. To address such a kind of problem, the scalarization method can be applied. To qualify the tradeoff, we incorporate a set of weight factors: $\{\alpha_1,\ \alpha_2,\alpha_3\}$, where $\alpha_1+\alpha_2+\alpha_3=1$, to reflect the relative importance of the energy costs, execution time and payment cost, respectively. For example, when the system is more energy constrained, then the weight factor α_1 can be made relatively larger and vice versa.

By such, the multi-objective optimization system is able to be transformed to a single objective optimization problem **P2**, which is

$$\min_{\{p_i^C,P_i\}}\quad \alpha_1\frac{E\left(p_i^C,P_i\right)}{\tilde{E}}+\alpha_2\frac{T\left(p_i^C,P_i\right)}{\tilde{T}}+\alpha_3\frac{M\left(p_i^C\right)}{\tilde{M}}. \tag{27}$$

subject to: (21)-(26).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

$$\begin{aligned} E\left(p_i^C, P_i\right) &= \frac{1}{N}\sum_{i=1}^{N} E_i\left(p_i^C, P_i\right) \\ &= \frac{1}{N}\left\{\sum_{i=1}^{N}\left[\left(1-p_i^C\right)E_i^M\left(p_i^C\right)+p_i^C E_i^S\left(p_i^C, P_i\right)\right]\right\} \\ &= \frac{1}{N}\left\{\sum_{i=1}^{N}\left[\left(1-p_i^C\right)\frac{\kappa_i}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i}+p_i^C\frac{P_i p_i^C \lambda_i \theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_i+\sum_{j\in N, j\neq i} P_j h_j}\right)}\right]\right\} \end{aligned} \tag{17}$$

$$\begin{aligned} &T\left(p_i^C, P_i\right) \\ &= \frac{1}{N}\left[\sum_{i=1}^{N} T_i\left(p_i^C, P_i\right)\right] \\ &= \frac{1}{N}\sum_{i=1}^{N}\left\{\left(1-p_i^C\right)T_i^M\left(p_i^C\right)+p_i^C\left[T_i^t\left(p_i^C, P_i\right)+\psi^F\left(T_{wait}^F\left(p_i^C\right)+T_b^F\left(p_i^C\right)\right)+\left(1-\psi^F\right)\left(T_{wait}^{CC}+T_b^{CC}\left(p_i^C\right)\right)\right]\right\} \\ &= \frac{1}{N}\sum_{i=1}^{N}\left\{\begin{array}{l}\left(1-p_i^C\right)\frac{1}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i}+p_i^C\frac{p_i^C\lambda_i\theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_0+\sum_{j\in N, j\neq i} P_j h_j}\right)} \\ +p_i^C\psi^F\left(\frac{C\left(c,\rho^F\right)}{cu^F-\lambda_p^F}+\frac{1}{u^F}+\frac{1}{u_b^F-\lambda_p^F}\right)+p_i^C\left(1-\psi^F\right)\left(T^O+\frac{1}{u^{CC}}+\frac{1}{u_b^{CC}-\left(\sum_{i=1}^{N}\lambda_i p_i^C-\lambda_p^F\right)}\right)\end{array}\right\} \end{aligned} \tag{18}$$

From the above description, we can see that the optimization variables p_i^C and P_i of MD $i, i \in \mathcal{N}$ interact with each other. When the value of p_i^C becomes larger, the energy consumption of the MD decreases while the execution delay increasing; when the value of P_i becomes larger, the energy consumption of the MD increase while the execution time decreases. Thus, in this paper, we should optimize the offloading probability p_i^C and transmission power P_i in order to optimize the energy consumption, execution time, and cost.

IV. Algorithm Design

Through a detailed analysis, we can find that the maximum request rate of the fog node is an important factor for the system performance as it can determine whether there is any requests transferred to the central cloud. By comparing the values of $\lambda_{\max}^F$ and λ_{Total}^M, we can further divided the case into two sub-cases.

1) In the first sub-case, we assume that the maximum request rate of the fog node is larger than the total workload in the system, i.e., $\lambda_{\max}^F \geq \lambda_{Total}^M$. In other words, all the MDs' requests in the system can be processed at the fog node. In this situation, ψ^F=1 and $\lambda_p^F = \lambda_{Total}^M = \sum_{i=1}^{N} p_i^C \lambda_i$. Substituting (1), (3), (4), (5), (10), (11), (12) and (19) into (27), we can obtain the E&D&P optimization problem **P3** in a specific analytical expression in (28) subject to

$$\left(1-p_i^C\right)\lambda_i < u_i^M\left(1-l_i^M\right), \tag{29}$$

$$\sum_{i=1}^{N}\lambda_i p_i^C - cu^F < 0, \tag{30}$$

$$\sum_{i=1}^{N}\lambda_i p_i^C - \lambda_{\max}^F < 0, \tag{31}$$

$$\sum_{i=1}^{N}\lambda_i p_i^C - u_b^F < 0, \tag{32}$$

$$0 \leq p_i^C \leq 1, \tag{33}$$

$$0 < P_i < P_i^{max}, \tag{34}$$

where $\rho^F = \frac{\lambda_p^F}{cu^F} = \frac{\sum_{i=1}^{N}\lambda_i p_i^C}{cu^F}$.

2) In the second subcase, we assume that the maximum requests rate of the fog node is less than the total requests workload in the system, which means that $\lambda_{\max}^F < \lambda_{Total}^M$. In this situation, $\psi^F = \frac{\lambda_{\max}^F}{\lambda_{Total}^M} = \frac{\lambda_{\max}^F}{\sum_{i=1}^{N} p_i^C \lambda_i}$. In other words, the fog node can only process as much as $\lambda_{\max}^F$ workload, and the overloaded requests will be further offloaded to the central cloud to execute. Substituting (1), (3), (4), (5), (10), (11), (12), (13), (14) and (19) into (27), we can obtain the E&D&P optimization problem **P4** in (35) where $\rho^F = \frac{\lambda_{\max}^F}{cu^F}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

$$\begin{aligned}
&\min_{\{p_i^C, P_i\}} \quad V_1\left(p_i^C, P_i\right) = \\
&\alpha_1 \frac{1}{N}\frac{1}{\tilde{E}}\sum_{i=1}^{N}\left[\left(1-p_i^C\right)\frac{\kappa_i}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i}+p_i^C\frac{P_i p_i^C \lambda_i \theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_0+\sum_{j\in N, j\neq i} P_j h_j}\right)}\right] \\
&+\alpha_2 \frac{1}{N}\frac{1}{\tilde{T}}\sum_{i=1}^{N}\left\{\begin{array}{l}\left(1-p_i^C\right)\frac{1}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i} \\ +p_i^C\left[\frac{p_i^C\lambda_i\theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_0+\sum_{j\in N, j\neq i} P_j h_j}\right)}+\frac{C\left(c,\rho^F\right)}{cu^F-\sum\limits_{i=1}^{N}\lambda_i p_i^C}+\frac{1}{u^F}+\frac{1}{u_b^F-\sum\limits_{i=1}^{N}\lambda_i p_i^C}\right]\end{array}\right\} \\
&+\alpha_3 r^F \frac{1}{N}\frac{1}{\tilde{M}}\sum_{i=1}^{N}\lambda_i p_i^C
\end{aligned} \tag{28}$$

$$\begin{aligned}
&\min_{\{p_i^C, P_i\}} \quad V_2\left(p_i^C, P_i\right) = \\
&\alpha_1 \frac{1}{N}\frac{1}{\tilde{E}}\left\{\sum_{i=1}^{N}\left[\left(1-p_i^C\right)\frac{\kappa_i}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i}+p_i^C\frac{P_i p_i^C \lambda_i \theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_0+\sum_{j\in N, j\neq i} P_j h_j}\right)}\right]\right\} \\
&+\alpha_2 \frac{1}{N}\frac{1}{\tilde{T}}\sum_{i=1}^{N}\left\{\begin{array}{l}\left(1-p_i^C\right)\frac{1}{u_i^M\left(1-l_i^M\right)-\left(1-p_i^C\right)\lambda_i}+p_i^C\frac{p_i^C\lambda_i\theta_i}{W\log_2\left(1+\frac{P_i h_i}{\omega_0+\sum_{j\in N, j\neq i} P_j h_j}\right)} \\ +p_i^C\left[\frac{\lambda_{\max}^F}{\sum\limits_{i=1}^{N}p_i^C\lambda_i}\left(\frac{C\left(c,\rho^F\right)}{cu^F-\lambda_{\max}^F}+\frac{1}{u^F}+\frac{1}{u_b^F-\lambda_{\max}^F}\right)\right] \\ +p_i^C\left(1-\frac{\lambda_{\max}^F}{\sum\limits_{i=1}^{N}p_i^C\lambda_i}\right)\left(T^O+\frac{1}{u^{CC}}+\frac{1}{u_b^{CC}-\left(\lambda_{Total}^M-\lambda_{\max}^F\right)}\right)\end{array}\right\} \\
&+\alpha_3 \frac{1}{N}\frac{1}{\tilde{M}}\left[r^F\lambda_{\max}^F+r^{CC}\left(\sum_{i=1}^{N}p_i^C\lambda_i-\lambda_{\max}^F\right)\right]
\end{aligned} \tag{35}$$

subject to

$$\left(1-p_i^C\right)\lambda_i < u_i^M\left(1-l_i^M\right), \tag{36}$$

$$\lambda_{\max}^F - \sum_{i=1}^{N}\lambda_i p_i^C < 0, \tag{37}$$

$$\sum_{i=1}^{N}\lambda_i p_i^C - \lambda_{\max}^F - u_b^{CC} < 0, \tag{38}$$

$$0 \le p_i^C \le 1, \tag{39}$$

$$0 < P_i < P_i^{max}. \tag{40}$$

In order to solve the nonlinear programming problems **P3** and **P4**, we may consider using one special "punishment" approach called IPM as presented in [**?**], [**?**]. The role of introducing penalty function is equivalent to setting obstacles on the boundary of the feasible region, so that the iterative solution process of solving always in the feasible region. Correspondingly, the penalty functions for the first subcase and second subcase are given as (41) and (42). In (41) and (42), $\xi_1^{(k)} > 0$ and $\xi_2^{(k)} > 0$ are the penalty coefficients, and $\xi_j^{(k)}, j \in \{1, 2\}$ satisfies the following iterative rules:

$$\xi_j^{(k+1)} = \beta_j \xi_j^{(k)}, \tag{43}$$

where β_j are the reduction factors. In general, the smaller of the reduction factor, the faster the penalty coefficient value falls, resulting in a larger interval of the optimal sequence. In contrast, the larger the reduction factor, the denser the interval of the optimal sequence, and the number of solving unconstrained optimal solution increases undoubtedly. In general, with $0 < \beta_j < 1$, we can always obtain the optimal solution.

Theorem 1: Assume that the feasible domain is a convex set and the constructed penalty function is continuous, then with the iterative IPM approach, at least one feasible solution can be obtained and it converges to the global optimum [24].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

$$\begin{aligned}
&\Phi_1\left(p_i^C, P_i, \xi_1^{(k)}\right) \\
&= V_1\left(p_i^C, P_i\right) - \xi_1^{(k)} \ln\left[\prod_{i=1}^{N}\left|\left(1-p_i^C\right)\lambda_i - u_i^M\left(1-l_i^M\right)\right|\right] - \xi_1^{(k)} \ln\left|\sum_{i=1}^{N}\lambda_i p_i^C - cu^F\right| \\
&- \xi_1^{(k)} \ln\left|\sum_{i=1}^{N}\lambda_i p_i^C - \lambda_{\max}^F\right| - \xi_1^{(k)} \ln\left|\sum_{i=1}^{N}\lambda_i p_i^C - u_b^F\right| - \xi_1^{(k)} \ln\left(\prod_{i=1}^{N}\left|p_i^C\right|\right) \\
&- \xi_1^{(k)} \ln\left(\prod_{i=1}^{N}\left|p_i^C - 1\right|\right) - \xi_1^{(k)} \ln\left(\prod_{i=1}^{N}\left|P_i\right|\right) - \xi_1^{(k)} \ln\left(\prod_{i=1}^{N}\left|P_i - P_i^{max}\right|\right)
\end{aligned} \tag{41}$$

$$\begin{aligned}
&\Phi_2\left(p_i^C, P_i, \xi_2^{(k)}\right) \\
&= V_2\left(p_i^C, P_i\right) - \xi_2^{(k)} \ln\left[\prod_{i=1}^{N}\left|\left(1-p_i^C\right)\lambda_i - u_i^M\left(1-l_i^M\right)\right|\right] - \xi_2^{(k)} \ln\left|\lambda_{\max}^F - \sum_{i=1}^{N}\lambda_i p_i^C\right| \\
&- \xi_2^{(k)} \ln\left|\sum_{i=1}^{N}\lambda_i p_i^C - \lambda_{\max}^F - u_b^{CC}\right| - \xi_2^{(k)} \ln\left(\prod_{i=1}^{N}\left|p_i^C\right|\right) - \xi_2^{(k)} \ln\left(\prod_{i=1}^{N}\left|p_i^C - 1\right|\right) \\
&- \xi_2^{(k)} \ln\left(\prod_{i=1}^{N}\left|P_i\right|\right) - \xi_2^{(k)} \ln\left(\prod_{i=1}^{N}\left|P_i - P_i^{max}\right|\right)
\end{aligned} \tag{42}$$

We can easily find that the constraints in **P3** and **P4** are linear, so the feasible domain are convex sets undeniably. Moreover, the penalty functions that we constructed in (41) and (42) are continuous. So we can obtain the global optimum with IPM approach as **Theorem 1** described.

By evaluating the following equations,

$$\begin{cases} \dfrac{\partial \Phi_1\left(p_i^C, P_i, \xi_1^{(k)}\right)}{\partial p_i^C} = 0, & (i = 1, 2, \cdots, N), \\ \dfrac{\partial \Phi_1\left(p_i^C, P_i, \xi_1^{(k)}\right)}{\partial P_i} = 0, & (i = 1, 2, \cdots, N). \end{cases} \tag{44}$$

$$\begin{cases} \dfrac{\partial \Phi_2\left(p_i^C, P_i, \xi_2^{(k)}\right)}{\partial p_i^C} = 0, & (i = 1, 2, \cdots, N), \\ \dfrac{\partial \Phi_2\left(p_i^C, P_i, \xi_2^{(k)}\right)}{\partial P_i} = 0, & (i = 1, 2, \cdots, N), \end{cases} \tag{45}$$

we can obtain the extreme points $\left(p_i^C\left(\xi_a^{(k)}\right), P_i\left(\xi_a^{(k)}\right)\right)_{i=1}^N$ of these two penalty functions. Through iteration, we can obtain the optimal solution $\left(\left(p_i^C\right)^*, \left(P_i\right)^*\right)_{i=1}^N$.

The detailed procedure of the proposed algorithm is depicted in Algorithm 1. With Algorithm 1, we can find the optimal offloading probability and the optimal transmit power for each MD in order to minimizing the E&D&P in the system under different cases.

The major advantages of the IPM scheme are the low-degree polynomial complexity, and an unrivalled ability to deliver optimal solutions in an almost constant number of iterations which depends very little, if at all, on the problem dimension. Thus, from a practical point of view, they have produced solutions to many industrial problems that were hitherto intractable. For the proposed IPM-based algorithm, the complexity is $\mathrm{O}(n)$ where n denotes the number of iterations [25].

Algorithm 1 Proposed IPM-based Algorithm

1: Initialization:
initial feasible point $\left(\left(p_i^C\right)^0, \left(P_i\right)^0\right)_{i=1}^N$; initial value of penalty coefficients $\xi_j^{(0)}$; the reduction factor β_j, $k = 0$.
2: Define ε_j as a sufficiently small positive real number.
3: Solving the extreme points of the penalty functions as $\left(p_i^C\left(\xi_j^{(k)}\right), P_i\left(\xi_j^{(k)}\right)\right)_{i=1}^N$ $(j = 1, 2)$.
4: **while** ($\left\|\left(\left(p_i^C\left(\xi_j^{(k)}\right), P_i\left(\xi_j^{(k)}\right)\right)_{i=1}^N\right) - \left(\left(\left(p_i^C\right)^0, \left(P_i\right)^0\right)_{i=1}^N\right)\right\| > \varepsilon_j$) **do**
5: Iteration: $\xi_j^{(k+1)} = \beta_j \xi_j^{(k)}$ $(j = 1, 2; k = 0, 1, 2 \cdots)$,
$\left(\left(p_i^C\right)^0, \left(P_i\right)^0\right)_{i=1}^N = \left(p_i^C\left(\xi_j^{(k)}\right), P_i\left(\xi_j^{(k)}\right)\right)_{i=1}^N$ $(j = 1, 2)$,$k = k + 1$.
6: **end while**
7: **return** $\left(p_i^C\left(\xi_j^{(k)}\right), P_i\left(\xi_j^{(k)}\right)\right)_{i=1}^N$ $(j = 1, 2)$.

TABLE II
SIMULATION PARAMETERS OF THE SINGLE-USER SCENARIO

Parameters	u^F (MIPS)	u_b^F (MIPS)	u_i^M (MIPS)	λ_i (MIPS)
Value	10	10	4.5	1.5
Parameters	κ_i (w)	θ_i (bits)	l_i^M	
Value	16	3.2e+6	0.3	

V. PERFORMANCE EVALUATIONS

In this section, extensive simulations are conducted to validate the effectiveness of the proposed algorithm for the joint E&D&P optimization problem. We also assume that the maximum energy consumption, delay and cost for MDs in the system is 15 Joule, 2 Second and 0.1, respectively. The unit payment using the fog computing is assumed to be 0.001 and for the central cloud is 0.005. The number of servers in the fog node is $c = 4$.

First, we investigate the impact of offloading probability

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

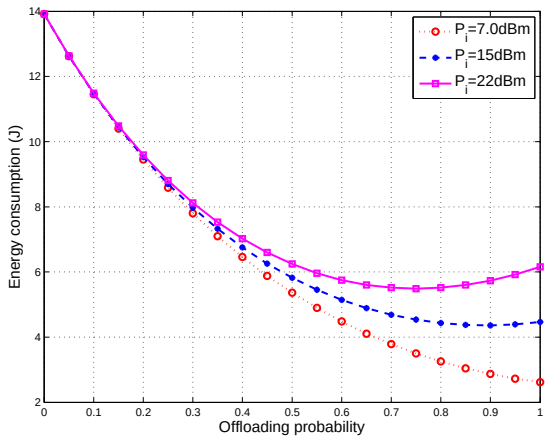


Fig. 2. The impact of offloading probability on energy consumption

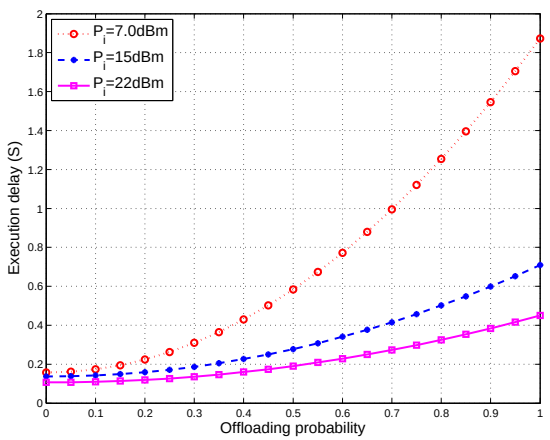


Fig. 3. The impact of offloading probability on execution delay

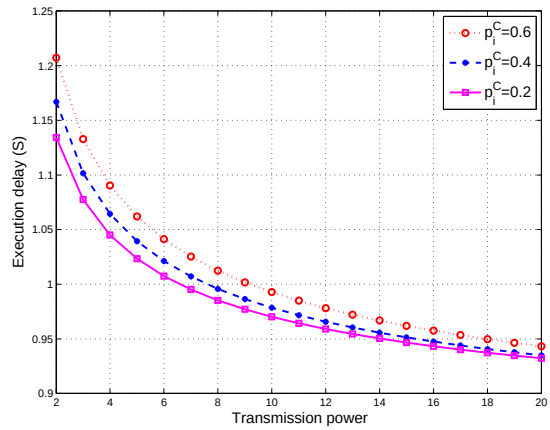


Fig. 4. The impact of transmission power on execution delay

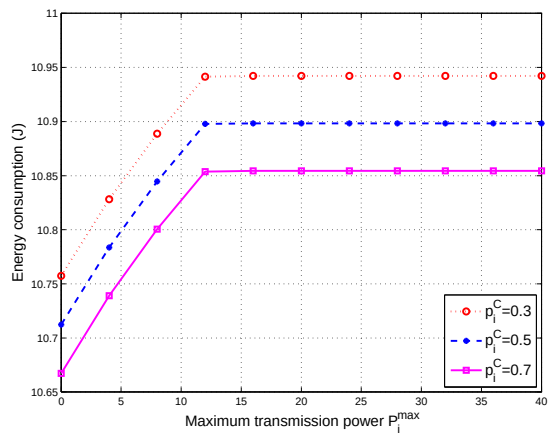


Fig. 5. The impact of maximum transmission power on execution delay

p_i^C and transmission power P_i on the energy consumption and delay performance. For simplicity, we concentrate on a single-user scenario. The simulation parameters can be found in Table II and some of them are modified from [8], [22]. In Fig. 2, we investigate the impact of offloading probability p_i^C on the energy consumption at different transmit powers. As we can see that at a certain transmit power, the energy consumption decreases with the increased offloading probability. When offloading probability increases, more and more requests are offloaded to the fog node. As considered, the transmit energy consumption is less than the local energy consumption, thus, the MD's energy consumption becomes less and less. From Fig. 2, the benefits on energy consumption of using MCC can be observed. Meanwhile, when the transmit power becomes larger, the transmission energy consumption also grows, which can be found by comparing the three curves at a certain offloading probability in Fig. 2.

In Fig. 3, the impact of offloading probability p_i^C on the execution delay at different transmit powers is illustrated. As we can see that at a certain level of transmit power, the execution delay increases along with the offloading probability. When more and more requests offloaded to the cloud, the transmission time and queue time will be increased, which is in line with the trend in Fig. 3 and also indicates the drawbacks of MCC on delay.

In Fig. 4, we also study the impact of transmit power P_i on the execution delay at different offloading probabilities. Generally, a larger transmits power can result in a larger uplink data rate, and obtain a smaller delay, which can be found from Fig. 4. Also, the lager of offloading probability, as more requests are offloaded to the cloud servers, the execution delay would naturally increase, which can be found by comparing the three curves in Fig. 4.

We examine the impact of maximum transmit power on energy consumption in Fig. 5. From this figure, we can find that at the beginning, the energy consumption increases with the increment of the maximum transmit power. The energy consumption approaches a constant value at a certain point. This is mainly because of the proposed optimization solution, the optimized transmit power level is reached. Comparing these four figures, we can clearly observe the necessity for investigating the tradeoff between the energy consumption and execution delay with respect to the offloading probability and transmit power.

Secondly, we evaluate the system performance of the first sub-case, where the fog node processing capability is relatively large, i.e., $\lambda_{Total}^M < \lambda_{\max}^F$ and assume that there are 3 MDs in the system if not specified. The simulation parameters of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

TABLE III
SIMULATION PARAMETERS OF THE FIRST-SUB CASE

Parameters (Units)	MD 1	MD 2	MD 3
κ_i (w)	16	16	16
θ_i (bits)	3.2e+6	2.7e+6	2.3e+6
u_i^M (MIPS)	4.6	4.5	4.5
P_i^{th} (dBm)	23	23	23
l_i^M	0.3	0.3	0.3

MDs for the first sub-case are presented in Table III, and they are also modified from [9] and [25].

With the proposed scheme, we can obtain the optimal offloading probability and optimal transmission power for each MD at any arrival rate at a certain weight set. For example, we determine $(\alpha_1, \alpha_2, \alpha_3) = (0.4, 0.5, 0.1)$, when the arrival rates are $(1.4, 1.8, 1.6)$, the optimal transmission power and optimal offloading probability is $(13.8976, 0.8021)$, $(10.5612, 0.7699)$, $(14.7653, 0.8357)$ for MD 1, MD 2, and MD 3 respectively.

We also investigate the optimal transmit power and optimal offloading probabilities for different sets of weight factors to see the impact of weight factors at a certain arrival rate for each MD In Table IV. We set arrival rate for each MD as $(1.4, 1.3, 1.6)$ respectively. The optimal transmission power and optimal offloading probability at different sets of weight factors for each MD is displayed at Table IV which illustrate the impact of weight factors. For example, when the system puts more attention on energy, the offloading probability is 0.8817 but the transmission power is 11.4533 for MD 1, which decrease the energy consumption from both higher offloading probability and lower transmit power, while the system puts more attention on delay performance, the the offloading probability is 0.5499 but the transmission power is 15.0433, which decrease the execution delay from both lower offloading probability and higher transmission power. When the MD puts more attention on the cost, the offloading probability is also relatively lower.

Then we evaluate the system performance of the second sub-case in Figs. 6- 8, where the processing capability of the fog node is smaller comparing with the requests, i.e., $\lambda_{Total}^M > \lambda_{\max}^F$. We assume that there are 10 MDs in the system unless specified.

With the proposed scheme, we can also compute the optimal offloading probability and optimal transmission power in order to minimize the system overhead at any arrival rates of the MDs at a certain weight set. For example, we determine $(\alpha_1, \alpha_2, \alpha_3) = (0.2, 0.4, 0.4)$, when the arrival rates are $(1.6, 2.1, 1.8)$ for MD 1, MD 2, and MD 3, the optimal transmission power and optimal offloading probability is $(12.2029, 0.8990)$, $(11.3717, 0.7308)$, $(11.4979, 0.8450)$, respectively.

In addition, we investigate the impact of the number of MDs on E&D&P, which is displayed in Fig. 6. From Fig. 6, we can find that when the number of MDs increases, the energy consumption and execution delay also increase, while the payment cost decreases. There is no doubt that resources contention and sharing can cause delay and performance degradation that might result in higher and higher response time. With the increased execution delay, some MDs prefer to execute some requests locally, so the energy consumption increases and payment cost decreases.

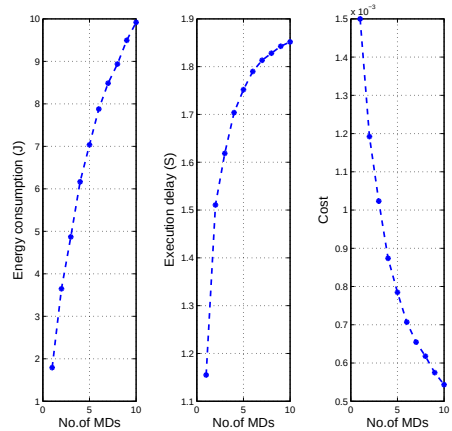


Fig. 6. The impact of number of MDs on E&D&P

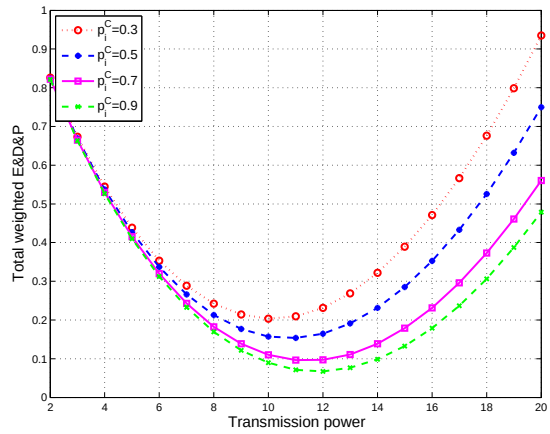


Fig. 7. The impact of transmission power on E&D&P

In Fig. 7, we investigate the impact of transmit power on the total E&D&P at different offloading probabilities. At first, with transmission power increasing, the total weighted E&D&P decrease. The E&D&P reaches the minimum, at a certain transmission power value, which is the optimal transmit power. Then the total weighted E&D&P increase with the transmit power increasing. This rule can be found from any curve in Fig.7, which denotes different offloading probabilities. Moreover, the larger offloading probability, the less E&D&P can be obtained, which can be found by comparing the four curves in Fig. 7.

In Fig. 8, we compare our proposed scheme with other schemes proposed in [14], [22]. In our scheme, we optimize both offloading probability and transmit power to minimize the E&D&P while the method in [14] can be viewed as the one only optimizes the offloading probability and the one in [22] only optimizes the transmit power. We can see that our method can achieve a better performance in E&D&P by jointly optimizing the offloading probability and transmit power, which demonstrates the comprehensiveness and validity of this study.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

TABLE IV
THE OPTIMAL TRANSMISSION POWER AND OPTIMAL OFFLOADING PROBABILITY OF THE FIRST SUB-CASE

Sets of weight factors	$((P_1)^*, (p_1^F)^*)$ of MD 1	$((P_2)^*, (p_2^F)^*)$ of MD 2	$((P_3)^*, (p_3^F)^*)$of MD 3
$(0.6, 0.2, 0.2)$	$(11.4533, 0.8817)$	$(10.4731, 0.8632)$	$(10.4680, 0.9237)$
$(0.2, 0.7, 0.1)$	$(15.0433, 0.5499)$	$(11.5344, 0.5354)$	$(11.5278, 0.5774)$
$(0.1, 0.2, 0.7)$	$(13.7332, 0.6246)$	$(11.1569, 0.5938)$	$(11.1201, 0.6439)$

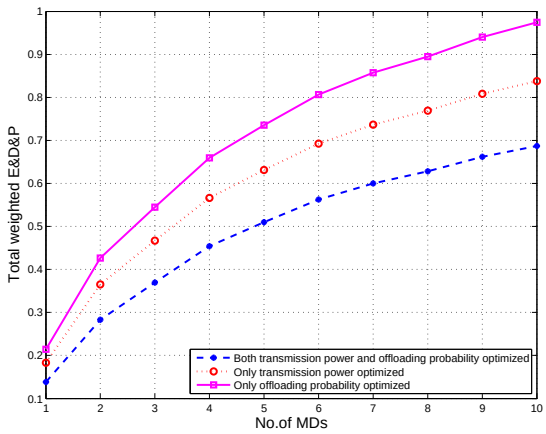


Fig. 8. Comparing among different schemes

VI. CONCLUSION

In this paper, we investigated the problem of energy consumption, delay performance and payment cost in a mobile fog computing system. Specifically, we optimized the offloading probability and transmission power for the MDs to jointly minimize the energy consumption, delay performance and cost. We derived analytic results on energy consumption, delay performance and payment cost assuming three different queueing models at mobile devices, the fog node and central cloud and explicit consideration of the wireless channel. By leveraging the obtained results, a multi-objective problem with various constraints is formulated and addressed by using an IPM-based algorithm. The performance evaluations were presented to illustrate the effectiveness of the proposed scheme and demonstrate the superior performance over the existing schemes.

REFERENCES

[1] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, and B. Li, "Gearing resource poor mobile devices with powerful clouds: architectures, challenges, and applications," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 14-22, Jun. 2013.

[2] M. Chiang and T. Zhang, "Fog and IoT: an Overview of research opportunities," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 854-864, Dec. 2016.

[3] Y. Jararweh, A. Doulat, O. AlQudah, E. Ahmed, M. Al-Ayyoub, and E. Benkhelifa, "The future of mobile cloud computing: integrating cloudlets and mobile edge computing," *in proceedings of 23rd International Conference on Telecommunications (ICT)*, Thessaloniki, Greece, May. 2016.

[4] W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu, "Edge computing: vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637-646, Oct. 2016.

[5] S. Deng, L. Huang, J. Taheri, and A. Y. Zomaya, "Computation offloading for service workflow in mobile cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 12, pp. 3317-3329, Dec. 2015.

[6] J. Song, Y. Cui, M. Li, J. Qiu, R. Buyya, "Energy-traffic tradeoff cooperative offloading for mobile cloud computing," *2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*, Hong Kong, China, May 2014.

[7] Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, "Heterogeneity in mobile cloud computing: taxonomy and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 369-392, Feb. 2014.

[8] S. Barbarossa, S. Sardellitti, and P. D. Lorenzo, "Communicating while computing: distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 45-55, Oct. 2014.

[9] X. Guo, L. Liu, Z. Chang, and T. Ristaniemi, "Data offloading and task allocation for cloudlet-assisted ad hoc mobile clouds," *Wireless Networks*, in press, Jun. 2016, DOI :10.1007/s11276-016-1322-z.

[10] N. Fernando, S. W. Loke, and W. Rahayu, "Computing with nearby mobile devices: a work sharing algorithm for mobile edge-clouds" *IEEE Transactions on Cloud Computing*, in press, Apr. 2016, DOI: 10.1109/TCC.2016.2560163.

[11] Y. Zhang, D. Niyato, and P. Wang, "Offloading in mobile cloudlet systems with intermittent connectivity," *IEEE Transactions on Mobile Computing*, vol. 14, no. 12, pp. 2516-2529, Dec. 2015.

[12] Y. C. Liu, and M. J. Lee, "Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system," *IEEE Transactions on Mobile Computing*, in press, Nov. 2015, DOI: 10.1109/TMC.2015.2504091.

[13] M. Jia, J. Cao, W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Transactions on Cloud Computing*, in press, Jun. 2015, DOI: 10.1109/TCC.2015.2449834.

[14] T. Zhao, S. Zhou, X. Guo, Y. Zhao, and Z. Niu, "A cooperative scheduling scheme of local cloud and internet cloud for delay-aware mobile cloud computing," *2015 IEEE Globecom Workshops (GC Wkshps)*, San Diego, CA, USA, Dec. 2015.

[15] X. Masip-Bruin, E. Marin-Tordera, G. Tashakor, A. Jukan and G. J. Ren, "Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems," *IEEE Wireless Communications*, vol. 23, no. 5, pp. 120-128, Oct. 2016.

[16] A. V. Dastjerdi and R. Buyya, "Fog computing: helping the internet of things realize its potential," *Computer*, vol. 49, no. 8, pp. 112-116, Aug. 2016.

[17] X. Chen, L. Jiao, W. Z. Li, and X. M. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computation," *IEEE/ACM Transactions on Networking*, in press, Oct. 2015, DOI: 10.1109/TNET.2015.2487344.

[18] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal workload allocation in fog-cloud computing towards balanced delay and power consumption," *IEEE Internet of Things Journal*, vol.3, no.6, pp. 1171 - 1181, Dec. 2016.

[19] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 2, pp. 89-103, Aug. 2015.

[20] Z. Jiang, and S. Mao, "Energy delay tradeoff in cloud offloading for multi-core mobile devices," *IEEE Access*, vol. 3, pp. 2306-2316, Nov. 2015.

[21] X. M. Wang, J. Wang, X. Wang, and X. M. Chen, "Energy and delay tradeoff for application offloading in mobile cloud computing," *IEEE Systems Journal*, in press, Aug. 2015, DOI: 10.1109/JSYST.2015.2466617.

[22] Y. Wang, X. Lin, M. Pedram, "A nested two stage game-based optimization framework in mobile cloud computing system,"*2013 IEEE Seventh International Symposium on Service-Oriented System Engineering*, Washington, USA, Mar. 2013.

[23] B. Ngo, and H. Lee, "Analysis of a pre-emptive priority M/M/c model with two types of customers and restriction," *Electronics Letters*, vol. 26, no. 15, pp. 1190-1192, Jul. 1990.

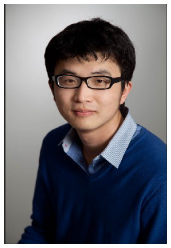
[24] D. Hu, Y. M. Alsmadi, and L. Y. Xu, "High-fidelity nonlinear IPM mod-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2780236, IEEE Internet of Things Journal

eling based on measured stator winding flux linkage," *IEEE Transactions on Industry Applications*, vol. 51, no. 4, pp. 3012-3019, Jul. 2015.

[25] J. Gondzio, "Interior point methods 25 years later," *European Journal of Operational Research*, vol. 218, no. 3, pp. 587-601, May 2012.

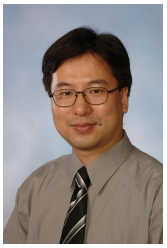
Liqing Liu received her master degree in College of Science at Yanshan University in 2015 and is now pursing for PhD degree in College of Information Science and Engineering at Yanshan University, Qinhuangdao, China. Her research interests include cloud computing and mobile computing.



Zheng Chang received the B.Eng. degree from Jilin University, Changchun, China, in 2007, the M.Sc. (Tech.) degree from the Helsinki University of Technology (Now Aalto University), Espoo, Finland, in 2009, and the Ph.D. degree from the University of Jyväskylä, Jyväskylä, Finland, in 2013. Since 2008, he has held various research positions at the Helsinki University of Technology, University of Jyväskylä and Magister Solutions Ltd., in Finland. He was a Visiting Researcher with Tsinghua University, China, in 2013, and the University of Houston, TX, USA, in 2015. He has been honored by the Ulla Tuominen Foundation, the Nokia Foundation, and the Riitta, Jorma J. Takanen Foundation, and the Jorma Ollila Grant for his research excellence. He is currently an Assistant Professor with the University of Jyväskylä.

He serves as an Editor of the IEEE Access, the Springer Wireless Networks, and the IEEE MMTC Communications Frontier, and a Guest Editor of the IEEE Access, the IEEE Communications Magazine, the IEEE Wireless Communications, the IEEE Internet of Things Journal, and the Wireless Communications and Mobile Computing. He has also served as a TPC member for many IEEE major conferences, such as Globecom, ICC, INFOCOM, PIMRC, and VTC. His research interests include IoT, cloud/edge computing, security and privacy, vehicular networks, and green communications.

Xijuan Guo received a PhD degree from Yanshan University. She is now a professor at College of Information Science and Engineering, Yanshan University, Qinhuangdao, China. Her research interests include high performance computing, cloud computing, image processing, wireless communications.



Shiwen Mao (S'99-M'04-SM'09) received Ph.D. in electrical and computer engineering from Polytechnic University, Brooklyn, NY. He is the Samuel Ginn Distinguished Professor, and Director of the Wireless Engineering Research and Education Center (WEREC) at Auburn University, Auburn, AL. His research interests include 5G wireless and IoT. He is a Distinguished Lecturer of the IEEE Vehicular Technology Society. He is on the Editorial Board of IEEE Transactions on Multimedia, IEEE Internet of Things Journal, IEEE Multimedia, ACM GetMobile, among others. He received the 2015 IEEE ComSoC TC-CSR Distinguished Service Award, the 2013 IEEE ComSoc MMTC Outstanding Leadership Award, and the NSF CAREER Award in 2010. He is a co-recipient of the Best Paper Awards from IEEE GLOBECOM 2016, IEEE GLOBECOM 2015, IEEE WCNC 2015, and IEEE ICC 2013, the Best Demo Award from IEEE SECON 2017, and the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems.

Tapani Ristaniemi received his M.Sc. in 1995 (Mathematics), Ph.Lic. in 1997 (Applied Mathematics) and Ph.D. in 2000 (Wireless Communications), all from the University of Jyväskylä, Jyväskylä, Finland. In 2001 he was appointed as Professor in the Department of Mathematical Information Technology, University of Jyväskylä. In 2004 he moved to the Department of Communications Engineering, Tampere University of Technology, Tampere, Finland, where he was appointed as Professor in Wireless Communications. In 2006 he moved back to University of Jyväskylä to take up his appointment as Professor in Computer Science. He is an Adjunct Professor of Tampere University of Technology. In 2013 he was a Visiting Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.

He has authored or co-authored over 150 publications in journals, conference proceedings and invited sessions. He served as a Guest Editor of IEEE Wireless Communications in 2011 and currently he is an Editorial Board Member of Wireless Networks and International Journal of Communication Systems. His research interests are in the areas of brain and communication signal processing and wireless communication systems research.